

# Set Similarity Search Beyond MinHash\*

Tobias Christiani

tobc@itu.dk

IT University of Copenhagen

Rasmus Pagh

pagh@itu.dk

IT University of Copenhagen

December 23, 2016

## Abstract

We present an improved data structure for approximate similarity (or “nearest neighbor”) search under Braun-Blanquet similarity. For  $\mathbf{x}, \mathbf{y} \subseteq \{1, \dots, d\}$ , Braun-Blanquet similarity is defined as  $B(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \cap \mathbf{y}| / \max(|\mathbf{x}|, |\mathbf{y}|)$ . Given a collection  $P$  of sets,  $(b_1, b_2)$ -approximate Braun-Blanquet similarity search asks to preprocess  $P$  such that given a query set  $\mathbf{q}$  for which there exists  $\mathbf{x} \in P$  with  $B(\mathbf{q}, \mathbf{x}) \geq b_1$ , we can efficiently return  $\mathbf{x}' \in P$  with  $B(\mathbf{q}, \mathbf{x}') > b_2$ .

Historically a different measure, the Jaccard similarity  $J(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \cap \mathbf{y}| / |\mathbf{x} \cup \mathbf{y}|$ , has been more widely studied. We argue that it suffices to study *regular* data structures in which the size of the query set  $\mathbf{q}$  and the sizes of sets in  $P$  are fixed. In this setting there is a 1-1 mapping between Jaccard similarity and Braun-Blanquet similarity, so the problems are equivalent. It follows from the seminal paper of Indyk and Motwani (STOC 1998) that the so-called *MinHash* locality-sensitive hash function can be used to construct a data structure for  $(j_1, j_2)$ -approximate Jaccard similarity with space  $O(n^{1+\rho} + dn)$  and query time  $O(dn^\rho)$ , where  $n = |P|$  and  $\rho = \log(1/j_1) / \log(1/j_2) < 1$ . In the regular setting our data structure improves  $\rho$  in the exponent to  $\rho' = \log\left(\frac{1+j_1}{2j_1}\right) / \log\left(\frac{1+j_2}{2j_2}\right)$  which is always strictly smaller than  $\rho$ . The exponent in terms of Braun-Blanquet similarity is  $\rho' = \log(1/b_1) / \log(1/b_2)$ .

Our main technical idea is to create a locality-sensitive mapping of a set  $\mathbf{x}$  to a set of memory locations by simulating a certain Galton-Watson branching process with pairwise independent choices. The algorithm is simple to describe and implement, and its analysis uses only elementary tools. Surprisingly, even though the locality-sensitive mapping is data-independent, for a large part of the parameter space  $\{(b_1, b_2) \mid 0 < b_2 < b_1 < 1\}$  our data structure outperforms the currently best data-*dependent* method by Andoni and Razenshteyn (STOC 2015).

On the lower bound side we show that any solution to the Braun-Blanquet similarity search problem based on a locality-sensitive mapping and with exponent strictly better than  $\rho' = \log(1/b_1) / \log(1/b_2)$  would break known lower bounds on locality-sensitive hashing. The lower bound is tight over the entire parameter space, and implies lower bounds for angular distance on the unit sphere that are very close to matching the best known upper bounds for much of the parameter space.

---

\*The research leading to these results has received funding from the European Research Council under the European Union’s 7th Framework Programme (FP7/2007-2013) / ERC grant agreement no. 614331.

# 1 Introduction

In this paper we consider the approximate set similarity problem or, equivalently, the problem of approximate Hamming near neighbor search in sparse vectors. Data that can be represented as sparse vectors is ubiquitous — a typical example is the representation of text documents as *term vectors*, where non-zero vector entries correspond to occurrences of words (or shingles). In order to perform identification of near-identical text documents in web-scale collections, Broder et al. [11, 12] designed and implemented *MinHash* (a.k.a. min-wise hashing), now understood as a locality-sensitive hash function [18]. This allowed approximate answers to similarity queries to be computed much faster than by other methods, and in particular made it possible to cluster the web pages of the AltaVista search engine (for the purpose of eliminating near-duplicate search results). Almost two decades after it was first described, MinHash remains one of the most widely used locality-sensitive hashing methods as witnessed by thousands of citations of [11, 12] as well as the ACM Paris Kanellakis Theory and Practice Award that Broder shared with Indyk and Charikar in 2012.

A *similarity measure* maps a pair of vectors to a similarity score in  $[0; 1]$ . It will often be convenient to interpret a vector  $\mathbf{x} \in \{0, 1\}^d$  as the set  $\{i \mid \mathbf{x}_i = 1\}$ . With this convention the *Jaccard similarity* of two vectors can be expressed as  $J(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \cap \mathbf{y}| / |\mathbf{x} \cup \mathbf{y}|$ . In *approximate similarity search* we are interested the problem of searching a data set  $P \subseteq \{0, 1\}^d$  for a vector of similarity at least  $j_1$  with a query vector  $\mathbf{q} \in \{0, 1\}^d$ , but allow the search algorithm to return a vector of similarity  $j_2 < j_1$ . To simplify the exposition we will assume throughout the introduction that all vectors are  $t$ -sparse, i.e., have the same Hamming weight  $t$ .

Recent theoretical advances in data structures for approximate *nearest neighbor* search in Hamming space [5] make it possible to beat the asymptotic performance of MinHash-based Jaccard similarity search (using the LSH framework of [18]) in cases where the similarity threshold  $j_2$  is not too small. However, numerical computations suggest that MinHash is always better when  $j_2 < 1/45$ .

In this paper we address the problem: Can similarity search using MinHash be improved *in general*? We give an affirmative answer by introducing CHOSEN PATH: a simple data-independent search method that strictly improves MinHash, and is always better than the data-dependent method of [5] when  $j_2 < 1/9$ . Similar to (data-independent) locality-sensitive filtering (LSF) methods [21, 15] our method works by mapping each data (or query) vector to a set of keys that must be stored (or looked up). The name stems from the way the mapping is constructed: As paths in a layered random graph where the vertices at each layer is identified with the set  $\{1, \dots, d\}$  of dimensions, and where a vector  $\mathbf{x}$  is only allowed to choose paths that stick to non-zero components  $\mathbf{x}_i$ . This is illustrated in Figure 1.

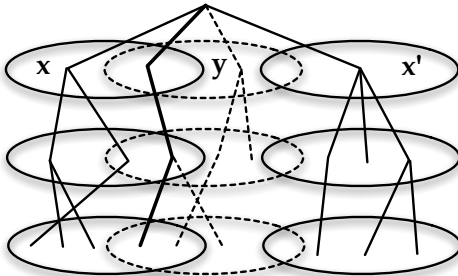


FIGURE 1: CHOSEN PATH uses a branching process to associate each vector  $\mathbf{x} \in \{0, 1\}^d$  with a set  $M_k(\mathbf{x}) \subseteq \{1, \dots, d\}^k$  of paths of length  $k$  (in the picture  $k = 3$ ). The paths associated with  $\mathbf{x}$  are limited to indices in the set  $\{i \mid \mathbf{x}_i = 1\}$ , represented by a circle at each level in the illustration. In the example the set sizes are:  $|M_3(\mathbf{x})| = |M_3(\mathbf{y})| = 4$  and  $|M_3(\mathbf{x}')| = 3$ . Parameters are chosen such that a query  $\mathbf{y}$  that is similar to  $\mathbf{x} \in P$  is likely to have a common path in  $M_k(\mathbf{x}) \cap M_k(\mathbf{y})$  (shown as a bold line), whereas it shares few paths in expectation with vectors such as  $\mathbf{x}'$  that are not similar.

## 1.1 Related work

High-dimensional approximate similarity search methods can be characterized in terms of their  $\rho$ -value which is the exponent for which queries can be answered in time  $O(n^\rho)$ , where  $n$  is the size of the set  $P$ . Here we focus on the “balanced” case where we aim for space  $O(n^{1+\rho})$ , but note that there now exist techniques for obtaining other trade-offs between query time and space overhead [4, 15]. We focus on results for Hamming space, which is a special case of similarity search on the unit sphere (many of the results cited apply to the more general case).

**Locality-sensitive hashing methods.** O’Donnell et al. [27] have shown that the value  $\rho = 1/c$  for  $c$ -approximate near neighbor search in Hamming space, obtained by in the seminal work of Indyk and Motwani [20], is the best possible in terms of  $c$  for schemes based on Locality-Sensitive Hashing (LSH). However, the lower bound only applies when the distances of interest,  $r$  and  $cr$ , are relatively small compared to  $d$ , and better bounds are known for large distances. Notably, Charikar’s SimHash [13] and cross-polytope LSH [2] give lower  $\rho$ -values for large distances. Extensions of the lower bound of [27] to cover more of the parameter space were recently given in [4, 15]. Until recently the best  $\rho$ -value known in terms of  $c$  was  $1/c$ , but in a sequence of papers Andoni et al. [3, 5] have shown how to use *data-dependent* LSH techniques to achieve  $\rho = 1/(2c - 1) + o_n(1)$ , bypassing the lower bound framework of [27] which assumes the LSH to be independent of data.

**Set similarity search.** Similarity search under set similarity, and the batched version often referred to as *set similarity join* [7, 8], has been studied extensively in the information retrieval and database literature, but mostly without providing theoretical guarantees on performance. Recently the notion of containment search, where the similarity measure is the (unnormalized) intersection size, was studied in the LSH framework [30]. This is a special case of *maximum inner product* search [30, 1]. However, these techniques do not give improvements in our setting.

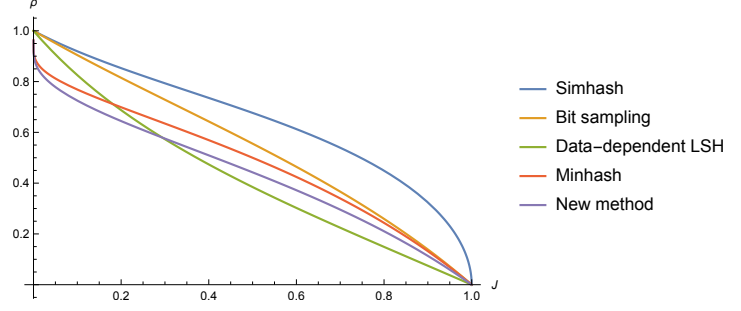
**Similarity estimation.** Finally, we mention that another application of MinHash [11, 12] is the (easier) problem of *similarity estimation*, where the task is to condense each vector  $\mathbf{x}$  into a short signature  $s(\mathbf{x})$  in such a way that the similarity  $J(\mathbf{x}, \mathbf{y})$  can be estimated from  $s(\mathbf{x})$  and  $s(\mathbf{y})$ . Thorup [32] has shown how to accomplish this using just a small amount of randomness in the definition of the function  $s(\cdot)$ . In another direction, Pham et al. [23] showed that it is possible to improve the performance of MinHash for similarity estimation when the Jaccard similarity is close to 1, but for smaller similarities it is known that succinct encodings of MinHash such as the one in [22] comes within a constant factor of the optimal space for storing  $s(\mathbf{x})$  [28]. Curiously, our improvement to MinHash in the context of similarity *search* comes when the similarity is neither too large or too small. Our techniques do not seem to yield any improvement for the similarity *estimation* problem.

## 1.2 Our contribution

In this paper we give the first strict improvement on the  $\rho$ -value for Jaccard similarity search compared to MinHash-based LSH. Figure 2 shows an example of the improvement for a slice of the parameter space. The improvement is based on a new locality-sensitive mapping that considers a specific random collection of length- $k$  paths on the vertex set  $\{1, \dots, d\}$ , and associates each vector  $\mathbf{x}$  with the paths in the collection that only visits vertices in  $\{i \mid \mathbf{x}_i = 1\}$ . Our data structure exploits that similar vectors will be associated with a common path with constant probability, while vectors with low similarity have a negligible probability of sharing a path.

However, the collection of paths has size superlinear in  $n$ , so an efficient method is required for locating the sets associated with a particular vector. Our choice of the collection of paths balances two opposing constraints: It is random enough to match the filtering performance of a truly random collection of sets, and at the same time it is structured enough to allow efficient search for sets matching a given vector. The search procedure is comparable in simplicity to the

FIGURE 2: Exponent when searching for a vector with Jaccard similarity  $J$  with approximation factor 2 (i.e., guaranteed to return a vector with Jaccard similarity  $J/2$ ) for various LSH methods. Our new method is the best data-independent method, and is better than data-dependent LSH up to about  $J \approx 0.3$ .



classical techniques of bit sampling, MinHash, SimHash, and  $p$ -stable LSH, and we believe it might be practical. This is in contrast to most theoretical advances in similarity search in the past ten years that suffer  $o(1)$  terms in the exponent of complexity bounds.

**Intuition.** Recall that we will think of a vector  $\mathbf{x} \in \{0, 1\}^d$  also as a set,  $\{i \mid \mathbf{x}_i = 1\}$ . Minhash can be thought of as a way of sampling an element  $i_{\mathbf{x}}$  from  $\mathbf{x}$ , namely, we let  $i_{\mathbf{x}} = \arg \min_{i \in \mathbf{x}} h(i)$  where  $h$  is a random hash function. For sets  $\mathbf{x}$  and  $\mathbf{y}$  the probability that  $i_{\mathbf{x}} = i_{\mathbf{y}}$  equals their Jaccard similarity  $J(\mathbf{x}, \mathbf{y})$ , which is much higher than if the samples had been picked independently. Consider the case in which  $|\mathbf{x}| = |\mathbf{y}| = t$ , so  $J(\mathbf{x}, \mathbf{y}) = \frac{|\mathbf{x} \cap \mathbf{y}|}{2t - |\mathbf{x} \cap \mathbf{y}|}$ . Another way of sampling is to compute  $I_{\mathbf{x}} = \mathbf{x} \cap \mathbf{b}$ , where  $\Pr[i \in \mathbf{b}] = 1/t$ , independently for each  $i \in [d]$ . The expected size of  $I_{\mathbf{x}}$  is 1, so this sample has the same expected “cost” as the MinHash-based sample. But if the Jaccard similarity is small, the latter samples are more likely to overlap:  $\Pr[I_{\mathbf{x}} \cap I_{\mathbf{y}}] = 1 - (1 - 1/t)^{|\mathbf{x} \cap \mathbf{y}|} \approx |\mathbf{x} \cap \mathbf{y}|/t$ , nearly a factor of 2 improvement. In fact  $\Pr[I_{\mathbf{x}} \cap I_{\mathbf{y}} \neq \emptyset] > \Pr[i_{\mathbf{x}} = i_{\mathbf{y}}]$  whenever  $|\mathbf{x} \cap \mathbf{y}| < 0.6t$ .

So in a certain sense, MinHash is not the best way of collecting evidence for the similarity of two sets. But intersection of the samples  $I_{\mathbf{x}}$  do not correspond directly to hash collisions, so it is not clear how to turn this insight into an algorithm in the LSH framework. Instead, we will use a variant of the locality-sensitive filtering (LSF) framework. It turns out that to most efficiently filter out vectors of low similarity, we should require not just that samples intersect but that they are *identical to b*. This leads to another obstacle: The family of samples needed to ensure this happens for similar vectors is very large. To overcome this we create the samples in a gradual, correlated way using a pairwise independent branching process that turns out to yield “sufficiently random” samples for the argument to go through.

## 2 Preliminaries

As stated above we will view  $\mathbf{x} \in \{0, 1\}^d$  both as a vector and as a subset of  $[d] = \{1, \dots, d\}$ . Define  $\mathbf{x}$  to be  $t$ -sparse if  $|\mathbf{x}| = t$ ; we will be interested in the setting where  $t \leq d/2$ , and typically the sparse setting  $t \ll d$ . Although many of the concepts we use hold for general spaces, for simplicity we state definitions in the same setting as our results: the boolean hypercube  $\{0, 1\}^d$  under some measure of similarity  $S: \{0, 1\}^d \times \{0, 1\}^d \rightarrow [0, 1]$ .

**Definition 1.** (Approximate similarity search) Let  $P \subset \{0, 1\}^d$  be a set of  $|P| = n$  data vectors, let  $S: \{0, 1\}^d \times \{0, 1\}^d \rightarrow [0, 1]$  be a similarity measure, and let  $s_1, s_2 \in [0, 1]$  such that  $s_1 > s_2$ . A solution to the  $(s_1, s_2)$ - $S$ -similarity problem is a data structure that supports the following query operation: on input  $\mathbf{q} \in \{0, 1\}^d$  for which there exists a vector  $\mathbf{x} \in P$  with  $S(\mathbf{x}, \mathbf{q}) \geq s_1$ , return  $\mathbf{x}' \in P$  with  $S(\mathbf{x}', \mathbf{q}) > s_2$ .

Our data structures will be randomized, and queries will succeed with probability at least  $1/2$  (which can be decreased arbitrarily by independent repetition). Sometimes similarity search is formulated as searching for vectors that are near  $\mathbf{q}$  according to the distance measure  $D(\mathbf{x}, \mathbf{y}) = 1 - S(\mathbf{x}, \mathbf{y})$ . For our purposes it is more natural to phrase conditions in terms of similarity, but we will compare to solutions originally described as “near neighbor” methods.

Many of the best known solutions to approximate similarity search problems are based on a technique of randomized space partitioning. This technique has been formalized in the locality-sensitive hashing framework [20] and the closely related locality-sensitive filtering framework [9, 15].

**Definition 2.** (Locality-Sensitive Hashing, LSH [20, 18]) A  $(s_1, s_2, p_1, p_2)$ -sensitive family of hash functions for a similarity measure  $S$  on  $\{0, 1\}^d$  is a distribution  $\mathcal{H}_S$  over functions  $h: \{0, 1\}^d \rightarrow R$  such that for all  $\mathbf{x}, \mathbf{y} \in \{0, 1\}^d$  and random  $h \in \mathcal{H}_S$ : If  $S(\mathbf{x}, \mathbf{y}) \geq s_1$  then  $\Pr[h(\mathbf{x}) = h(\mathbf{y})] \geq p_1$ , and if  $S(\mathbf{x}, \mathbf{y}) \leq s_2$  then  $\Pr[h(\mathbf{x}) = h(\mathbf{y})] \leq p_2$ .

The range  $R$  of the family will typically be fairly small such that an element of  $R$  can be represented in a constant number of machine words. In the following we further assume for simplicity that the family is *efficient* such that evaluation of a function can be done in time  $O(d)$ .

Given such a family it is quite simple to obtain a solution to the approximate similarity search problem, essentially by hashing points to buckets such that close points end up in the same bucket while distant points are kept apart.

**Lemma 1** ([20, 18]). *Given a  $(s_1, s_2, p_1, p_2)$ -sensitive family of hash functions with evaluation time  $O(d)$  it is possible to solve the  $(s_1, s_2)$ - $S$ -similarity problem with query time  $O(dn^\rho)$  and space usage  $O(dn + n^{1+\rho})$  where  $\rho = \log(1/p_1)/\log(1/p_2)$ .*

The upper bound presented in this paper does not quite fit into the existing frameworks. However, we would like to apply existing LSH lower bound techniques to our algorithm. Therefore we define a more general framework that captures solutions constructed using the LSH and LSF framework, as well as the upper bound presented in this paper.

**Definition 3** (Locality-Sensitive Maps). A  $(s_1, s_2, m_1, m_2)$ -sensitive family of maps for a similarity measure  $S$  on  $\{0, 1\}^d$  is a distribution  $\mathcal{M}_S$  over mappings  $M: \{0, 1\}^d \rightarrow 2^R$  (where  $2^R$  denotes the power set of  $R$ ) such that for all  $\mathbf{x}, \mathbf{y} \in \{0, 1\}^d$  and random  $M \in \mathcal{M}_S$ :

1.  $\mathbb{E}[|M(\mathbf{x})|] \leq m_1$ .
2. If  $S(\mathbf{x}, \mathbf{y}) \leq s_2$  then  $\mathbb{E}[|M(\mathbf{x}) \cap M(\mathbf{y})|] \leq m_2$ .
3. If  $S(\mathbf{x}, \mathbf{y}) \geq s_1$  then  $\Pr[M(\mathbf{x}) \cap M(\mathbf{y}) \neq \emptyset] \geq 1/2$ .

Once we have a family of locality-sensitive maps  $\mathcal{M}$  we can use it to obtain a solution to the  $(s_1, s_2)$ - $S$ -similarity problem.

**Lemma 2.** *Given a  $(s_1, s_2, m_1, m_2)$ -sensitive family of maps  $\mathcal{M}$  we can solve the  $(s_1, s_2)$ - $S$ -similarity problem with query time  $O(m_1 + nm_2|\mathbf{q}| + T_M)$  and space usage  $O(nm_1 + \sum_{\mathbf{x} \in P} |\mathbf{x}|)$  where  $T_M$  is the time to evaluate a map  $M \in \mathcal{M}$ .*

*Proof.* We construct the data structure by sampling a map  $M$  from  $\mathcal{M}$  and use it to place points in  $P$  into buckets. To run a query for a point  $\mathbf{q}$  we proceed by evaluating  $M(\mathbf{q})$  and computing the similarity between  $\mathbf{q}$  and the points in the buckets associated with  $M(\mathbf{q})$ . If a sufficiently similar point is found we return it. The running time guarantees result from repetition and applying Markov's inequality.  $\square$

**Model of computation.** We assume the standard word RAM model [17] with word size  $\Theta(\log n)$ , where  $n = |P|$ . In order to be able to draw random functions from a family of functions we augment the model with an instruction that generates a machine word uniformly at random in constant time.

### 3 Upper bound

We will describe a family of locality-sensitive maps  $\mathcal{M}_B$  for solving the  $(b_1, b_2)$ - $B$ -similarity problem where

$$B(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \cap \mathbf{y}| / \max(|\mathbf{x}|, |\mathbf{y}|) \quad (1)$$

is a similarity measure due to Braun-Blanquet [10]. After describing  $\mathcal{M}_B$  we will give an efficient implementation of  $M \in \mathcal{M}_B$  and show that we can set parameters to obtain our upper bound:

**Theorem 1.** *For every choice of constants  $0 < b_2 < b_1 < 1$  we can solve the  $(b_1, b_2)$ - $B$ -similarity problem with query time  $O(|\mathbf{q}|n^\rho \log n)$  and space usage  $O(n^{1+\rho} \log n + \sum_{\mathbf{x} \in P} |\mathbf{x}|)$  where  $\rho = \log(1/b_1) / \log(1/b_2)$ .*

#### 3.1 Chosen Path

The CHOSEN PATH family  $\mathcal{M}_B$  is defined by  $k$  random hash functions  $h_1, \dots, h_k$  where  $h_i: [w] \times [d]^i \rightarrow [0; 1]$  and  $w$  is a positive integer. The evaluation of a map  $M_k \in \mathcal{M}_B$  proceeds in a sequence of  $k + 1$  steps that can be analyzed as a Galton-Watson branching process, originally devised to investigate population growth under the assumption of identical and independent offspring distributions. In the first step  $i = 0$  we create a population of  $w$  starting points

$$M_0(\mathbf{x}) = [w]. \quad (2)$$

In subsequent steps, every path that has survived so far produces offspring according to a random process that depends on  $h_i$  and the element  $\mathbf{x} \in \{0, 1\}^d$  being evaluated. We use  $p \circ j$  to denote concatenation of a path  $p$  with a vertex  $j$ .

$$M_i(\mathbf{x}) = \left\{ p \circ j \mid p \in M_{i-1}(\mathbf{x}) \wedge h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|} \right\}. \quad (3)$$

Observe that  $h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|}$  can only hold when  $\mathbf{x}_j = 1$ , so the paths in  $M_i(\mathbf{x})$  are constrained to  $j \in \mathbf{x}$ . The set  $M(\mathbf{x}) = M_k(\mathbf{x})$  is given by the paths that survive to the  $k$ th step. We will proceed by bounding the evaluation time of  $M \in \mathcal{M}_B$  as well as showing the locality-sensitive properties of  $\mathcal{M}_B$ . In particular, for similar points  $\mathbf{x}, \mathbf{y} \in \{0, 1\}^d$  with  $B(\mathbf{x}, \mathbf{y}) \geq b_1$  we will show that with probability at least  $1/2$  there will be a path that is chosen by both  $\mathbf{x}$  and  $\mathbf{y}$ .

**Lemma 3** (Properties of CHOSEN PATH). *For all  $\mathbf{x}, \mathbf{y} \in \{0, 1\}^d$  and random  $M \in \mathcal{M}_B$ :*

1.  $\mathbb{E}[|M_i(\mathbf{x})|] \leq (1/b_1)^i w$ .
2. If  $B(\mathbf{x}, \mathbf{y}) < b_2$  then  $\mathbb{E}[|M_i(\mathbf{x}) \cap M_i(\mathbf{y})|] \leq (b_2/b_1)^i w$ .
3. If  $B(\mathbf{x}, \mathbf{y}) \geq b_1$  then  $\Pr[M_i(\mathbf{x}) \cap M_i(\mathbf{y}) \neq \emptyset] \geq 1 - i/w$ .

*Proof.* We prove each property by induction on  $i$ . The base cases  $i = 0$  follow from (2). Now consider the inductive step for property 1. Let  $\mathbb{1}\{\mathcal{P}\}$  denote the indicator function for predicate  $\mathcal{P}$ . Using independence of the hash functions  $h_i$  we get:

$$\begin{aligned} \mathbb{E}[|M_i(\mathbf{x})|] &= \mathbb{E} \left[ \sum_{p \in M_{i-1}(\mathbf{x})} \sum_{j \in [d]} \mathbb{1} \left\{ h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|} \right\} \right] \\ &= \mathbb{E} \left[ \sum_{p \in M_{i-1}(\mathbf{x})} 1 \right] \mathbb{E} \left[ \sum_{j \in [d]} \mathbb{1} \left\{ h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|} \right\} \right] \\ &\leq \mathbb{E}[|M_{i-1}(\mathbf{x})|] / b_1 \\ &\leq (1/b_1)^i w. \end{aligned}$$

The last inequality uses the induction hypothesis. We use the same approach for the second property where we let  $X_i = M_i(\mathbf{x}) \cap M_i(\mathbf{y})$ .

$$\begin{aligned}
\mathbb{E}[X_i] &= \mathbb{E} \left[ \sum_{p \in X_{i-1}} \sum_{j \in [d]} \mathbb{1} \left\{ h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|} \wedge h_i(p \circ j) < \frac{\mathbf{y}_j}{b_1 |\mathbf{y}|} \right\} \right] \\
&= \mathbb{E} \left[ \sum_{p \in X_{i-1}} 1 \right] \sum_{j \in [d]} \Pr \left[ h_i(p \circ j) < \frac{\min(\mathbf{x}_j, \mathbf{y}_j)}{b_1 \max(|\mathbf{x}|, |\mathbf{y}|)} \right] \\
&\leq \mathbb{E}[|X_{i-1}|] (B(\mathbf{x}, \mathbf{y})/b_1) \\
&\leq (B(\mathbf{x}, \mathbf{y})/b_1)^i w.
\end{aligned}$$

To prove the third property we bound the variance of  $|X_i|$  and apply Chebyshev's inequality to bound the probability of  $X_i = \emptyset$ . First consider the case where  $|\mathbf{x}| \leq 1/b_1$  and  $|\mathbf{y}| \leq 1/b_1$ . Here it must hold that  $X_i > 0$  as intersecting paths exist ( $b_1 > 0$ ) and always activate. In all other cases we have that  $\mathbb{E}[|X_i|] = (B(\mathbf{x}, \mathbf{y})/b_1)^i w$ . Knowing the expected value we can apply Chebyshev's inequality once we have an upper bound for  $\text{Var}[|X_i|] = \mathbb{E}[|X_i|^2] - \mathbb{E}[|X_i|]^2$ . Specifically we show that  $\mathbb{E}[|X_i|^2] \leq wi$ , by induction on  $i$ . To simplify notation we define the following indicator variable (suppressing the subscript  $i$ ).

$$Y_{p,j} = \mathbb{1} \left\{ h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|} \wedge h_i(p \circ j) < \frac{\mathbf{y}_j}{b_1 |\mathbf{y}|} \right\}.$$

We have  $\mathbb{E}[Y_{p,j}] = |\mathbf{x} \cap \mathbf{y}| / (b_1 \max(|\mathbf{x}|, |\mathbf{y}|))$  and by (3)  $|X_i| = \sum_{p \in X_{i-1}} \sum_{j \in [d]} Y_{p,j}$ , which means:

$$\begin{aligned}
\mathbb{E}[|X_i|^2] &= \mathbb{E} \left[ \left( \sum_{p \in X_{i-1}} \sum_{j \in [d]} Y_{p,j} \right)^2 \right] \\
&= \mathbb{E} \left[ \sum_{p \in X_{i-1}} \sum_{j \in \mathbf{x} \cap \mathbf{y}} Y_{p,j}^2 \right] + \mathbb{E} \left[ \sum_{p, p' \in X_{i-1}} \sum_{j, j' \in \mathbf{x} \cap \mathbf{y}} Y_{p,j} Y_{p',j'} \mathbb{1}\{(p, j) \neq (p', j')\} \right] \\
&< \mathbb{E}[|X_{i-1}|] \cdot |\mathbf{x} \cap \mathbf{y}| / (b_1 \max(|\mathbf{x}|, |\mathbf{y}|)) + \mathbb{E}[|X_{i-1}|^2] \cdot (|\mathbf{x} \cap \mathbf{y}| / (b_1 \max(|\mathbf{x}|, |\mathbf{y}|)))^2 \\
&\leq w + \mathbb{E}[|X_{i-1}|^2] \\
&\leq w + w(i-1) \\
&\leq wi.
\end{aligned}$$

Here we use that  $|\mathbf{x} \cap \mathbf{y}| / (b_1 \max(|\mathbf{x}|, |\mathbf{y}|)) \leq w$  and  $\mathbb{E}[|X_i|] = 1$  by assumption on  $B(\mathbf{x}, \mathbf{y})$ . The third property now follows from Chebychev's inequality applied to  $|X_i|$ .<sup>1</sup>  $\square$

### 3.2 Implementation details and parameter settings

Lemma 3 continues to hold even when the hash functions  $h_1, \dots, h_k$  are individually *2-independent* (and mutually independent) since we only use bounds on the first and second moment of the hash values. We can therefore use a simple and practical scheme such as Zobrist hashing [34] that hashes strings of  $O(\log n)$  bits to strings of  $O(\log n)$  bits in  $O(1)$  time using space, say,  $O(\sqrt{n})$ . It is not hard to see that the domain and range of  $h_1, \dots, h_k$  can be compressed to using  $O(\log n)$  bits (causing a negligible increase in the failure probability of the data structure). We simply hash the paths  $p \in M_i(\mathbf{x})$  to intermediate values of  $O(\log n)$  bits, avoiding collisions with high probability, and in a similar vein, with high probability  $O(\log n)$  bits of precision suffice to determine whether  $h_i(p \circ j) < \frac{\mathbf{x}_j}{b_1 |\mathbf{x}|}$ .

<sup>1</sup>We note that a slightly better bound of probability  $1 - i/(2w)$  can also be shown.

We show how to parameterize  $\mathcal{M}_B$  to solve the  $(b_1, b_2)$ - $B$ -similarity problem on a set  $P$  of  $|P| = n$  points for every choice of constant parameters  $0 < b_2 < b_1 < 1$ , independent of  $n$ . Note that we exclude  $b_1 = 1$  (which would correspond to identical vectors that can be found in time  $O(1)$  by resorting to standard hashing) and  $b_2 = 0$  (for which every data point would be a valid answer to a query). We set parameters

$$k = \lceil \log(n) / \log(1/b_2) \rceil,$$

$$w = 4k$$

from which it follows directly that  $\mathcal{M}_B$  is  $(b_1, b_2, m_1, m_2)$ -sensitive with  $m_1 = n^\rho w / b_1$  and  $m_2 = n^{\rho-1} w$  where  $\rho = \log(1/b_1) / \log(1/b_2)$ . To bound the expected evaluation time of  $M_k$  we use Zobrist hashing as well as intermediate hashes for the paths as described above. In the  $i$ th step in the branching process the expected number of hash function evaluations is bounded by  $|\mathbf{q}|$  times the number of paths alive at step  $i - 1$ . We can therefore bound the expected time to compute  $M_k(\mathbf{q})$  by

$$\sum_{i=0}^{k-1} \mathbb{E}[|\mathbf{q}| |M_i(\mathbf{q})|] \leq \frac{b_1^{-k} - 1}{b_1^{-1} - 1} |\mathbf{q}| w = O(|\mathbf{q}| n^\rho w).$$

This completes the proof of Theorem 1.<sup>2</sup>

### 3.3 A solution for Jaccard similarity

We can use our solution for the Braun-Blanquet similarity problem from Theorem 1 to solve the approximate similarity search problem under Jaccard similarity  $J(\mathbf{x}, \mathbf{y}) = |\mathbf{x} \cap \mathbf{y}| / |\mathbf{x} \cup \mathbf{y}|$ . The solution we present here is a theoretical proof-of-concept where we have aimed for simplicity in the exposition at the cost of  $\text{poly log } n$  factors (hidden in the  $\tilde{O}$ -notation) in the query time and space overhead. The details behind the techniques used here are covered in greater detail and for more general similarity measures in Appendix A.

**Theorem 2.** *For every choice of constants  $0 < j_2 < j_1 < 1$  we can solve the  $(j_1, j_2)$ - $J$ -similarity problem with query time  $\tilde{O}(|\mathbf{q}| + n^\rho)$  and space usage  $\tilde{O}(n^{1+\rho})$  where  $\rho = \log\left(\frac{2j_1}{1+j_1}\right) / \log\left(\frac{2j_2}{1+j_2}\right)$ .*

As shown in Appendix A we can use standard techniques to reduce the universe size to  $\hat{d} = \text{poly log } n$  such that for every pair of points in the original  $d$ -dimensional universe it holds with high probability in  $n$  that we incur no more than an additive error of  $O(1/\log n)$  when computing their Jaccard similarity in the reduced universe. We can therefore solve the  $(j_1, j_2)$ - $J$ -similarity problem in  $\{0, 1\}^{\hat{d}}$  by solving the  $(j_1 + \varepsilon, j_2 - \varepsilon)$ - $J$ -similarity problem in  $\{0, 1\}^{\hat{d}'}$  where  $\varepsilon = O(1/\log n)$ . This results in no more than a  $\text{poly log } n$  factor increase in query time and space usage, and we can therefore ignore it for the remainder of the argument.

Let  $\mathbf{x}, \mathbf{y} \in \{0, 1\}^{\hat{d}}$  denote elements of the original universe and let  $\hat{\mathbf{x}}, \hat{\mathbf{y}} \in \{0, 1\}^{\hat{d}}$  denote their counterparts in the reduced universe. We can now proceed by creating  $(\hat{d} + 1)^2 = \text{poly log } n$  different data structures: one for each  $(t, t')$ -regular problem where  $t, t' \in [\hat{d} + 1]$  and  $t$  denotes the Hamming weight of data points while  $t'$  denotes the Hamming weight of query points. A data point  $\hat{\mathbf{x}}$  is stored in each of the  $\hat{d} + 1$  data structures for the  $(|\hat{\mathbf{x}}|, t')$ -regular problem. Similarly, a query for a point  $\hat{\mathbf{q}}$  is performed on each of the  $\hat{d} + 1$  data structures for the  $(t, |\hat{\mathbf{q}}|)$ -regular problem.

We use the Braun-Blanquet solution of Theorem 1 for each of the data structures for the  $(t, t')$ -regular problems. Knowing the weights of data and query points we can set parameters

<sup>2</sup>We know of a way of replacing the multiplicative factor  $|\mathbf{q}|$  by an additive term of  $O(|\mathbf{q}| \log |\mathbf{q}|)$  by choosing the hash functions  $h_i$  carefully, but do not discuss this improvement here since  $|\mathbf{q}|$  can be assumed to be polylogarithmic and our focus is on the exponent of  $n$ .



Measure Ref.	Hamming $r_1 < r_2$	Braun-Blanquet $b_1 > b_2$	Jaccard $j_1 > j_2$
Bit-sampling LSH [18]	$r_1/r_2$	$\frac{1-b_1}{1-b_2}$	$\frac{1-j_1}{1+j_1} / \frac{1-j_2}{1+j_2}$
Minhash LSH [11]	$\log \frac{1-r_1}{1+r_1} / \log \frac{1-r_2}{1+r_2}$	$\log \frac{b_1}{2-b_1} / \log \frac{b_2}{2-b_2}$	$\log(j_1) / \log(j_2)$
Angular LSH [2]	$\frac{r_1}{r_2} \frac{1-r_2/2}{1-r_1/2}$	$\frac{1-b_1}{1+b_1} / \frac{1-b_2}{1+b_2}$	$\frac{1-j_1}{1+3j_1} / \frac{1-j_2}{1+3j_2}$
Data-dep. LSH [5]	$\frac{r_1}{r_2} \frac{1}{2-r_1/r_2}$	$\frac{1-b_1}{1+b_1-2b_2}$	$\frac{(1-j_1)(1+j_2)}{1-j_1j_2+3(j_1-j_2)}$
<b>Theorem 1</b>	$\log(1-r_1)/\log(1-r_2)$	$\log(b_1)/\log(b_2)$	$\log \frac{2j_1}{1+j_1} / \log \frac{2j_2}{1+j_2}$

FIGURE 3: Overview of  $\rho$ -values for similarity search with Hamming vectors of equal weight  $t$ , in terms of various similarity measures. While most results in the literature are stated for a single measure, the fixed weight restriction gives a 1-1 correspondence that makes it possible to express the results in terms of other similarity measures. Hamming distances are normalized by a factor  $2t$  to lie in  $[0, 1]$ . Lower order terms of  $\rho$ -values are suppressed and for bit-sampling LSH we assume that the Hamming distance is small relative to the dimensionality of the space  $2r_1t/d = o(1)$  which holds for sparse data i.e.  $t/d = o(1)$ .

$b_1, b_2$  to reflect our target Jaccard similarity thresholds  $j_1, j_2$  by using the following relation between the two measures of similarity:

$$B(\mathbf{x}, \mathbf{y}) = \frac{|\mathbf{x}| + |\mathbf{y}|}{\max(|\mathbf{x}|, |\mathbf{y}|)} \frac{J(\mathbf{x}, \mathbf{y})}{1 + J(\mathbf{x}, \mathbf{y})}.$$

For Jaccard similarity thresholds  $j_1, j_2$  we therefore get Braun-Blanquet similarity thresholds for a data structure for the  $(t, t')$ -regular problem of  $b_1 = \gamma \cdot j_1 / (1 + j_1)$  and  $b_2 = \gamma \cdot j_2 / (1 + j_2)$  where  $\gamma = (t + t') / \max(t, t')$  is some number in  $[1, 2]$ . By analyzing the function  $\rho = \log(b_1) / \log(b_2) = \log(\gamma \cdot j_1 / (1 + j_1)) / \log(\gamma \cdot j_2 / (1 + j_2))$  we see that the  $\rho$ -value is maximized when  $\gamma = 2$  which means that the space usage and query time is dominated by  $(t, t')$ -regular data structures where  $t = t'$ , resulting in the  $\rho$ -value of Theorem 2.

### 3.4 Comparison

We will proceed by comparing our Theorem 1 to results that can be achieved using existing techniques. Again we focus on the setting where data points and query points are exactly  $t$ -sparse. In Appendix A we further show that for a large class of set similarity measures, including Braun-Blanquet and Jaccard similarity, the result of the comparison for the  $t$ -sparse setting continues to hold when we remove the assumption of fixed sparsity, up to a cost of  $\text{poly} \log n$  factors in the upper bound. We will only care about polynomial differences, ignoring the overhead factors of size  $n^{o(1)}$  that are present in many methods. An overview of different techniques for three measures of similarity is shown in Figure 3. To summarize: The CHOSEN PATH algorithm of Theorem 1 improves upon all existing data-independent results over the entire  $0 < b_2 < b_1 < 1$  parameter space. Furthermore, we improve upon the best known *data-dependent* techniques [5] for a large part of the parameter space (see Figure 6). The details of the comparisons are given in Appendix C.

**MinHash.** Since all vectors are  $t$ -sparse we have  $J(\mathbf{x}, \mathbf{y}) = B(\mathbf{x}, \mathbf{y}) / (2 - B(\mathbf{x}, \mathbf{y}))$ , that is, there is a 1-1 mapping between the two measures. Let  $b_1 = 2j_1 / (j_1 + 1)$  and  $b_2 = 2j_2 / (j_2 + 1)$

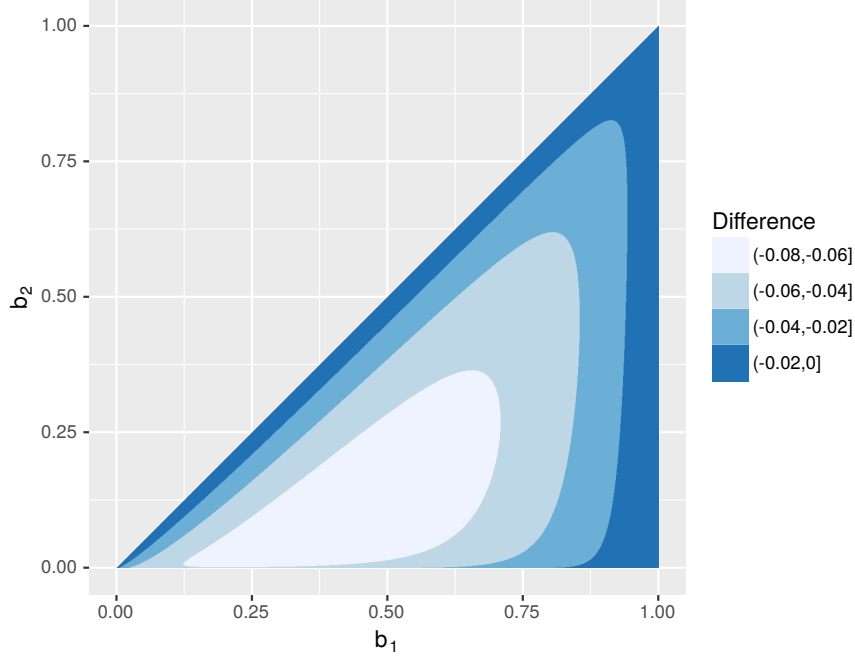


FIGURE 4: The difference  $\rho - \rho_{\text{minhash}}$  comparing CHOSEN PATH and MinHash in terms of Braun-Blanquet similarities  $0 < b_2 < b_1 < 1$ .

be the Braun-Blanquet similarities corresponding to Jaccard similarities  $j_1$  and  $j_2$ . The LSH framework using MinHash achieves  $\rho_{\text{minhash}} = \log\left(\frac{b_1}{2-b_1}\right) / \log\left(\frac{b_2}{2-b_2}\right)$ ; this should be compared to  $\rho = \log(b_1) / \log(b_2)$  achieved in Theorem 1. Since the function  $f(z) = \log(\frac{z}{2-z}) / \log z$  is monotonely increasing in  $[0; 1]$  we have that  $\rho / \rho_{\text{minhash}} = f(b_2) / f(b_1) < 1$ , i.e.,  $\rho$  is always smaller than  $\rho_{\text{minhash}}$ . As an example, for  $j_1 = 0.2$  and  $j_2 = 0.1$  we get  $\rho = 0.644\dots$  while  $\rho_{\text{minhash}} = 0.698\dots$ . Figure 4 shows the difference for the whole parameter space.

**Angular LSH.** Since our vectors are exactly  $t$ -sparse Braun-Blanquet similarities correspond directly to dot products (which in turn correspond to angles). Thus we can apply angular LSH such as SimHash [13] or cross-polytope LSH [2]. As observed in [15] one can express the  $\rho$ -value of cross-polytope LSH in terms of dot products as  $\rho_{\text{angular}} = \frac{1-b_1}{1+b_1} / \frac{1-b_2}{1+b_2}$ . Since the function  $f'(z) = (1+z) \log(z) / (1-z)$  is negative and monotonely increasing in  $[0; 1]$  we have that  $\rho / \rho_{\text{angular}} = f'(b_1) / f'(b_2) < 1$ , i.e.,  $\rho$  is always smaller than  $\rho_{\text{angular}}$ . In the above example, for  $j_1 = 0.2$  and  $j_2 = 0.1$  we have  $\rho_{\text{angular}} = 0.722\dots$  which is about 0.078 more than CHOSEN PATH. See figure 5 for a visualization of the difference for the whole parameter space.

**Data-dependent Hamming nearest neighbor.** The Hamming distance between two  $t$ -sparse vectors with Braun-Blanquet similarity  $b$  is  $2t(1-b)$ , since the intersection of the vectors has size  $tb$ . This means that  $(b_1, b_2)$ - $B$ -similarity search can be reduced to Hamming similarity search with approximation factor  $c = (2t(1-b_1)) / (2t(1-b_2)) = (1-b_1) / (1-b_2)$ . As mentioned above, the *data dependent* LSH technique of [5] achieves  $\rho = 1 / (2c - 1)$  ignoring  $o_n(1)$  terms. In terms of  $b_1$  and  $b_2$  this is  $\rho_{\text{datadep}} = \frac{1-b_1}{1+b_1-2b_2}$ , which is incomparable to the  $\rho$  of Theorem 1. In Appendix C we show that  $\rho < \rho_{\text{datadep}}$  whenever  $b_2 \leq 1/5$ , or equivalently, whenever  $j_2 \leq 1/9$ . Revisiting the above example, for  $j_1 = 0.2$  and  $j_2 = 0.1$  we have  $\rho_{\text{datadep}} = 0.6875$  which is about 0.043 more than CHOSEN PATH. Figure 6 gives a comparison covering the whole parameter space.

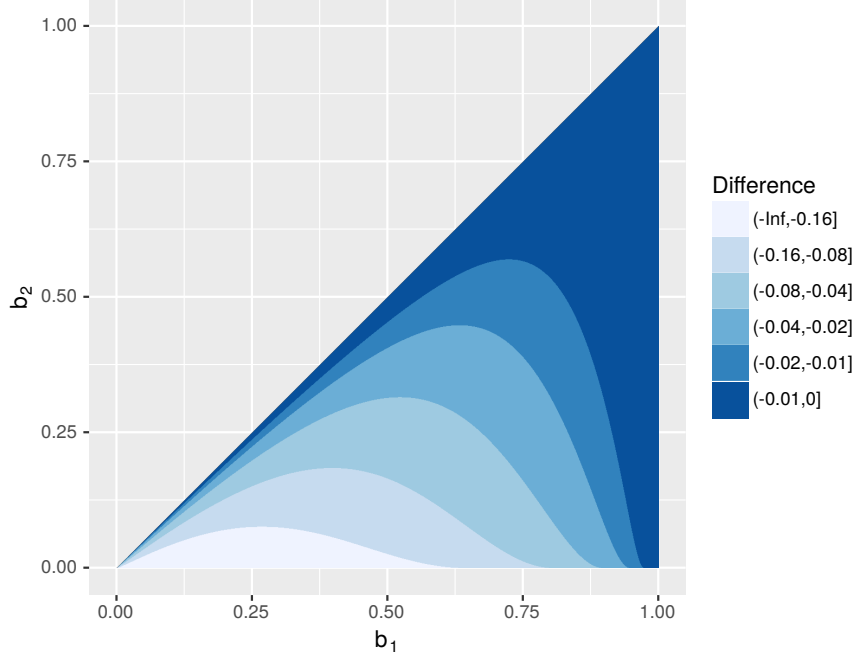


FIGURE 5: The difference  $\rho - \rho_{\text{angular}}$  comparing CHOSEN PATH and angular LSH in terms of Braun-Blanquet similarities  $0 < b_2 < b_1 < 1$ .

## 4 Lower bound

In this section we will show a locality-sensitive hashing lower bound for  $\{0, 1\}^d$  under Braun-Blanquet similarity. We will first show that LSH lower bounds apply to the class of solutions to the approximate similarity search problem that are based on locality-sensitive maps, thereby including our own upper bound. Next we will introduce some relevant tools from the literature, in particular the LSH lower bounds for Hamming space by O’Donnell et al. [27] which we use, through a reduction, to show LSH lower bounds under Braun-Blanquet similarity.

**Lower bounds for locality-sensitive maps.** Because our upper bound is based on a locality-sensitive map  $\mathcal{M}_B$  and not LSH-based we first show that LSH lower bounds apply to LSM-based solutions. This is not too surprising as both the LSH and LSF frameworks produce LSM-based solutions. We note that the idea of showing lower bounds for a more general class of algorithms that encompasses both LSH and LSF was used by Andoni et al. [4] in their list-of-points data structure lower bound for the space-time tradeoff of solutions to the approximate near neighbor problem in the random data regime. We use the approach of Christiani [15] to convert an LSM family into an LSH family using MinHash.

**Lemma 4.** *Suppose we have a  $(s_1, s_2, m_1, m_2)$ -sensitive family of maps  $\mathcal{M}$ . Then we can construct a  $(s_1, s_2, p_1, p_2)$ -sensitive family of hash functions  $\mathcal{H}$  with  $p_1 = 1/8m$  and  $q = m_2/m$  where  $m = \lceil 8m_1 \rceil$ .*

*Proof.* We sample a function  $h$  from  $\mathcal{H}$  by sampling a function  $M$  from  $\mathcal{M}$ , modify  $M$  to output a set of fixed size, and apply MinHash to the resulting set. For  $M \in \mathcal{M}$  we define the function  $\tilde{M}$  where we ensure that the size of the output set is  $m$ . We note that the purpose of this step is to be able to simultaneously lower bound  $p_1$  and upper bound  $p_2$  for  $\mathcal{H}$  when we apply MinHash to the resulting sets.

$$\tilde{M}(\mathbf{x}) = \begin{cases} \{(\mathbf{x}, 1), (\mathbf{x}, 2), \dots, (\mathbf{x}, m)\} & \text{if } |M(\mathbf{x})| \geq m, \\ \{(\mathbf{x}, 1), (\mathbf{x}, 2), \dots, (\mathbf{x}, m - |M(\mathbf{x})|)\} \cup M(\mathbf{x}) & \text{otherwise.} \end{cases}$$

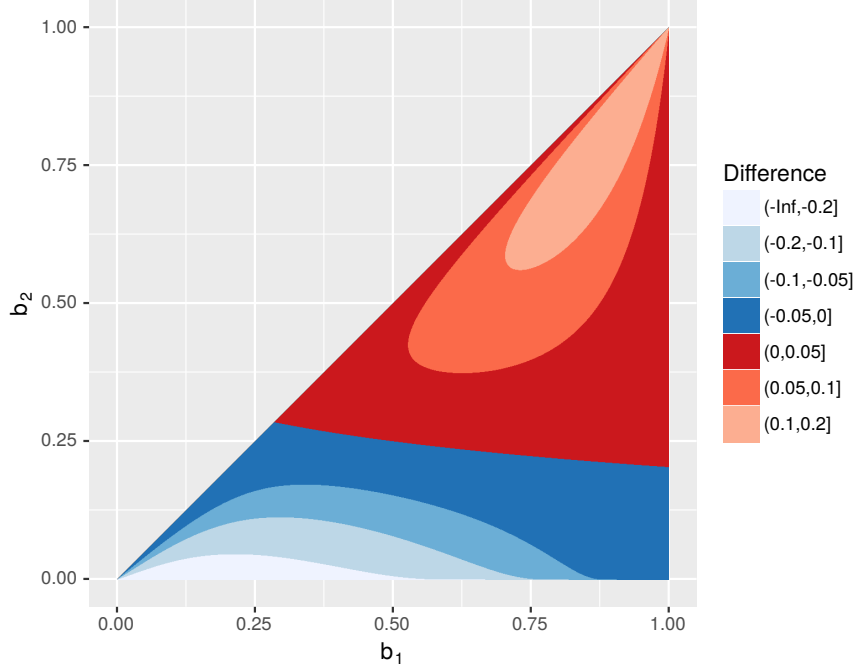


FIGURE 6: The difference  $\rho - \rho_{\text{datadep}}$  comparing CHOSEN PATH and data-dependent LSH in terms of Braun-Blanquet similarities  $0 < b_2 < b_1 < 1$ . In the area of the parameter space that is colored blue we have that  $\rho \leq \rho_{\text{datadep}}$  while for the red area it holds that  $\rho > \rho_{\text{datadep}}$ .

We proceed by applying MinHash. Let  $\pi$  denote a random permutation of the elements of the range of  $\tilde{M}$  and define

$$h(\mathbf{x}) = \min\{\pi(\tilde{M}(\mathbf{x}))\}.$$

We then have

$$\Pr[h(\mathbf{x}) = h(\mathbf{y})] = \sum_{\xi} \Pr[J(\tilde{M}(\mathbf{x}), \tilde{M}(\mathbf{y})) = \xi] \cdot \xi$$

summing over the finite set of all possible Jaccard similarities  $\xi = a/b$  with  $a, b \in \{0, 1, \dots, 2m\}$ . It is now fairly simple to lower bound  $p_1$  and upper bound  $p_2$ . We let  $\mathbf{x}, \mathbf{y}$  denote close pairs of points with  $S(\mathbf{x}, \mathbf{y}) \geq s_1$  and  $\mathbf{x}, \mathbf{x}'$  denote distant points with  $S(\mathbf{x}, \mathbf{x}') < s_2$ . Making use of the fact that  $m \leq |\tilde{M}(\mathbf{x}) \cup \tilde{M}(\mathbf{y})| \leq 2m$

$$\begin{aligned} \Pr[\tilde{M}(\mathbf{x}) \cap \tilde{M}(\mathbf{y}) = \emptyset] &\leq \Pr[M(\mathbf{x}) \cap M(\mathbf{y}) = \emptyset \wedge |M(\mathbf{x})| \geq m \wedge |M(\mathbf{y})| \geq m] \\ &\leq \Pr[M(\mathbf{x}) \cap M(\mathbf{y}) = \emptyset] + \Pr[|M(\mathbf{x})| \geq m] + \Pr[|M(\mathbf{y})| \geq m] \\ &\leq 1/2 + 1/8 + 1/8 \\ &= 3/4 \end{aligned}$$

Where we use the properties of  $\mathcal{M}$  together with Markov's inequality to upper bound the probabilities. We therefore have that  $\Pr[\tilde{M}(\mathbf{x}) \cap \tilde{M}(\mathbf{y}) \neq \emptyset] \geq 1/4$  and the lower bound on  $p_1$  follows. For distant pairs of points we get

$$\sum_{\xi} \Pr[J(\tilde{M}(\mathbf{x}), \tilde{M}(\mathbf{x}')) = \xi] \cdot \xi \leq (1/m) \sum_{i=1}^{\infty} \Pr[|\tilde{M}(\mathbf{x}) \cup \tilde{M}(\mathbf{x}')| \geq i] \cdot i = \frac{m_2}{m}.$$

□

We are now ready to justify the statement that LSH lower bounds apply to LSM, allowing us to restrict our attention to proving LSH lower bounds for Braun-Blanquet similarity.

**Corollary 1.** *Suppose that we have an LSM-based solution to the  $(s_1, s_2)$ -S-similarity problem with query time  $O(n^\rho)$ . Then there exists a family  $\mathcal{H}$  of locality-sensitive hash functions with  $\rho(\mathcal{H}) = \rho + O(1/\log n)$ .*

*Proof.* The existence of the LSM-based solution implies that for every  $n$  there exists a  $(s_1, s_2, m_1, m_2)$ -sensitive family of maps  $\mathcal{M}$  with  $m_1, nm_2 = O(n^\rho)$ . The upper bound on  $\rho$  follows from applying Lemma 4.  $\square$

**Overview of LSH lower bounds for Hamming space.** There exists a number of powerful results that lower bound the  $\rho$ -value that can be attained through locality-sensitive hashing and related approaches in various settings [25, 29, 27, 6, 15, 4]. O’Donnell et al. [27] showed an LSH lower bound of  $\rho = \log(1/p_1)/\log(1/p_2) \geq 1/c - o_d(1)$  for Hamming space under the assumption that  $\log(1/p_2) = o(d)$ . Their lower bound holds for  $(r, cr, p_1, p_2)$ -sensitive families for a particular choice of  $r$  that depends on  $d, p_2$ , and  $c$ , and where  $r$  is small compared to  $d$  (for instance, we have that  $r = \tilde{\Theta}(d^{2/3})$  when  $c$  and  $p_2$  are constant). We state a simplified version of the O’Donnell et al. lower bound where  $r = \sqrt{d}$ . The lower bound is weaker in the sense that it only holds for a more narrow range of  $p_2$ , however this does not matter in the proof of our lower bound for Braun-Blanquet similarity. The full proof of Lemma 5 is given in Appendix B.

**Lemma 5.** *For every  $d \in \mathbb{N}$ ,  $1/d \leq p_2 \leq 1 - 1/d$ , and  $1 \leq c \leq d^{1/8}$  every  $(\sqrt{d}, c\sqrt{d}, p_1, p_2)$ -sensitive hash family  $\mathcal{H}$  for  $\{0, 1\}^d$  under Hamming distance must have*

$$\rho(\mathcal{H}) = \frac{\log(1/p_1)}{\log(1/p_2)} \geq \frac{1}{c} - O(d^{-1/4}). \quad (4)$$

In general, good lower bounds for the entire parameter space  $(r, cr)$  are not known, although the techniques by O’Donnell et al. appear to yield a bound of  $\rho \gtrsim \log(1 - 2r/d)/\log(1 - 2cr/d)$ . This is far from tight as can be seen by comparing it to the bit-sampling [20] upper bound of  $\rho = \log(1 - r/d)/\log(1 - cr/d)$ . Existing lower bounds are tight in two different settings. Firstly, in the setting where  $cr \approx d/2$  (random data), lower bounds [25, 16, 6] match the various instantiations of spherical LSH [31, 3, 2]. Secondly, in the setting where  $r \ll d$ , the lower bound by O’Donnell et al. [27] becomes  $\rho \gtrsim \log(1 - 2r/d)/\log(1 - 2cr/d) \approx 1/c$ , matching bit-sampling LSH [20] as well as spherical LSH.

#### 4.1 LSH lower bound for Braun-Blanquet similarity

We are now ready to state our lower bound.

**Theorem 3.** *For every choice of constants  $0 < b_2 < b_1 < 1$  every  $(b_1, b_2, p_1, p_2)$ -sensitive hash family  $\mathcal{H}_B$  for  $\{0, 1\}^d$  under Braun-Blanquet similarity must satisfy*

$$\rho(\mathcal{H}_B) = \frac{\log(1/p_1)}{\log(1/p_2)} \geq \frac{\log(1/b_1)}{\log(1/b_2)} - O\left(\frac{\log(d/q)}{d}\right)^{1/3}. \quad (5)$$

*Remark.* The lower bound together with Corollary 1 shows that our solutions for Braun-Blanquet and Jaccard similarity in Theorem 1 and 2 are optimal up to  $o(1)$  terms in the exponent. Furthermore, the lower bound also applies to angular distance on the unit sphere where it comes close to matching the best known upper bounds for much of the parameter space as can be seen from Figure 5.

The proof works by assuming the existence of a  $(b_1, b_2, p_1, p_2)$ -sensitive family  $\mathcal{H}_B$  for  $\{0, 1\}^d$  under Braun-Blanquet similarity with  $\rho = \log(1/b_1)/\log(1/b_2) - \gamma$  for some  $\gamma > 0$ . We use a transformation  $T$  from Hamming space to Braun-Blanquet similarity to show that the existence of  $\mathcal{H}_B$  implies the existence of a  $(r, cr, p'_1, p'_2)$ -sensitive family  $\mathcal{H}_H$  for  $D$ -dimensional Hamming space that will contradict the lower bound of O’Donnell et al. [27] as stated in Lemma 5 for some appropriate choice of  $\gamma = \gamma(d, p_2)$ .

**From Hamming distance to Braun-Blanquet similarity.** Let  $d \in \mathbb{N}$  and let  $0 < b_2 < b_1 < 1$  be constant as in Theorem 3. Let  $\varepsilon \geq 1/d$  be a parameter to be determined. We want to show how to use a transformation  $T: \{0,1\}^D \rightarrow \{0,1\}^d$  from Hamming distance to Braun-Blanquet similarity together with our family  $\mathcal{H}_B$  to construct a  $(r, cr, p'_1, p'_2)$ -sensitive family  $\mathcal{H}_H$  for  $D$ -dimensional Hamming space with parameters

$$\begin{aligned} D &= 2^d \\ r &= \sqrt{D} \\ c &= \frac{\ln(1/(b_2 - \varepsilon))}{\ln(1/(b_1 + \varepsilon))} \end{aligned}$$

where  $p'_1$  and  $p'_2$  remain to be determined.

The function  $T$  takes as parameters positive integers  $t$ ,  $l$ , and  $\tau$ . The output of  $T$  consists of  $t$  concatenated  $l$ -bit strings, each of Hamming weight one. Each of the  $t$  strings is constructed independently at random according to the following process: Sample a vector of indices  $\mathbf{i} = (i_1, i_2, \dots, i_\tau)$  uniformly at random from  $[D]^\tau$  and define  $\mathbf{x}_{\mathbf{i}} \in \{0,1\}^\tau$  as  $\mathbf{x}_{\mathbf{i}} = \mathbf{x}_{i_1} \circ \mathbf{x}_{i_2} \circ \dots \circ \mathbf{x}_{i_\tau}$ . Let  $\mathbf{z}(\mathbf{x}) \in \{0,1\}^{2^\tau}$  be indexed by  $j \in \{0,1\}^\tau$  and set the bits of  $\mathbf{z}(\mathbf{x})$  as follows:

$$\mathbf{z}(\mathbf{x})_j = \begin{cases} 1 & \text{if } \mathbf{x}_{\mathbf{i}} = j, \\ 0 & \text{otherwise.} \end{cases}$$

Next we apply a random function  $g: \{0,1\}^\tau \rightarrow [l]$  in order to map  $\mathbf{z}(\mathbf{x})$  down to an  $l$ -bit string  $\mathbf{r}(\mathbf{z}(\mathbf{x}))$  of Hamming weight one while approximately preserving Braun-Blanquet similarity. For  $i \in [l]$  we set

$$\mathbf{r}(\mathbf{z}(\mathbf{x}))_i = \bigvee_{j: g(j)=i} \mathbf{z}(\mathbf{x})_j.$$

Finally we set

$$T(\mathbf{x}) = \mathbf{r}_1(\mathbf{z}_1(\mathbf{x})) \circ \mathbf{r}_2(\mathbf{z}_2(\mathbf{x})) \circ \dots \circ \mathbf{r}_t(\mathbf{z}_t(\mathbf{x}))$$

where for  $i \in [t]$  we construct each  $\mathbf{r}_i(\mathbf{z}_i(\mathbf{x}))$  independently.

We state the properties of  $T$  for the following parameter setting:

$$\begin{aligned} \tau &= \lfloor \sqrt{D} \ln(1/(b_1 + \varepsilon)) \rfloor \\ l &= \lceil 8/\varepsilon \rceil \\ t &= \lfloor d/l \rfloor. \end{aligned}$$

**Lemma 6.** *There exists a distribution over functions of the form  $T: \{0,1\}^D \rightarrow \{0,1\}^d$  such that for all  $\mathbf{x}, \mathbf{y} \in \{0,1\}^D$  and random  $T$ :*

1.  $|T(\mathbf{x})| = t$ .
2. If  $\|\mathbf{x} - \mathbf{y}\|_1 \leq r$  then  $B(T(\mathbf{x}), T(\mathbf{y})) \geq b_1$  with probability at least  $1 - e^{-t\varepsilon^2/2}$ .
3. If  $\|\mathbf{x} - \mathbf{y}\|_1 > cr$  then  $B(T(\mathbf{x}), T(\mathbf{y})) < b_2$  with probability at least  $1 - 2e^{t\varepsilon^2/32}$ .

*Proof.* The first property is trivial. For the second property we consider  $\mathbf{x}, \mathbf{y}$  with  $\|\mathbf{x} - \mathbf{y}\|_1 \leq r$  where we would like to lower bound

$$B(T(\mathbf{x}), T(\mathbf{y})) = \frac{|T(\mathbf{x}) \cap T(\mathbf{y})|}{\max(|T(\mathbf{x})|, |T(\mathbf{y})|)}.$$

We know that  $|T(\mathbf{x})| = t$  so it remains to lower bound the size of the intersection  $|T(\mathbf{x}) \cap T(\mathbf{y})|$ . Consider the expectation

$$\mathbb{E}[|T(\mathbf{x}) \cap T(\mathbf{y})|] = t \Pr[\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{y})].$$

We have that  $\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{y})$  if  $\mathbf{x}$  and  $\mathbf{y}$  take on the same value in the  $\tau$  bits that we sample. Under the assumption that  $\varepsilon \geq 1/d$ , then for  $d$  greater than some sufficiently large constant we can use standard approximations to the exponential function to show that

$$\begin{aligned}\Pr[\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{y})] &\geq (1 - r/D)^\tau \\ &\geq (1 - 1/\sqrt{D})^{\sqrt{D} \ln(1/(b_1 + \varepsilon))} \\ &\geq e^{\ln(b_1 + \varepsilon)} (1 - (\ln(b_1 + \varepsilon))^2 / \sqrt{D}) \\ &\geq b_1 + \varepsilon/2.\end{aligned}$$

Seeing as  $|T(\mathbf{x}) \cap T(\mathbf{y})|$  is the sum of  $t$  independent Bernoulli trials we can apply Hoeffding's inequality to yield the following bound:

$$\Pr[|T(\mathbf{x}) \cap T(\mathbf{y})| \leq b_1 t] \leq e^{-t\varepsilon^2/2}.$$

This proves the second property of  $T$  as listed above.

For the third property we consider the Braun-Blanquet similarity of distant pairs of points  $\mathbf{x}, \mathbf{x}'$  with  $\|\mathbf{x} - \mathbf{x}'\|_1 > cr$ . Again, under our assumption that  $\varepsilon \geq 1/d$  and for  $d$  greater than some constant we have

$$\begin{aligned}\Pr[\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{x}')] &\leq (1 - cr/D)^\tau \\ &\leq \left(1 - \frac{c}{\sqrt{D}}\right)^{-1} \left(1 - \frac{\ln(1/(b_2 - \varepsilon))}{\sqrt{D} \ln(1/(b_1 + \varepsilon))}\right)^{\sqrt{D} \ln(1/(b_1 + \varepsilon))} \\ &\leq (1 + 2c/\sqrt{D})(b_2 - \varepsilon) \\ &\leq b_2 - \varepsilon/2.\end{aligned}$$

There are two things that can cause the event  $B(T(\mathbf{x}), T(\mathbf{x}')) < b_2$  to fail. Firstly, the sum of the  $t$  independent Bernoulli trials for the event  $\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{x}')$  can deviate too much from its expected value. Secondly, the mapping down to  $l$ -bit strings that takes place from  $\mathbf{z}(\mathbf{x})$  to  $\mathbf{r}(\mathbf{z}(\mathbf{x}))$  can lead to an additional increase in the similarity due to collisions. Let  $Z$  denote the sum of the  $t$  Bernoulli trials for the events  $\mathbf{z}(\mathbf{x}) = \mathbf{z}(\mathbf{x}')$  associated with  $T$ . We again apply a standard Hoeffding bound to show that

$$\Pr[Z \geq (b_2 - \varepsilon/4)t] \leq e^{-t\varepsilon^2/8}.$$

Let  $X$  denote the number of collisions when performing the universe reduction to  $l$ -bit strings. By our choice of  $l$  we have that  $E[X] \leq (\varepsilon/8)t$ . Another application of Hoeffding's inequality shows that

$$\Pr[X \geq (\varepsilon/4)t] \leq e^{-t\varepsilon^2/32}.$$

We therefore get that

$$\Pr[|T(\mathbf{x}) \cap T(\mathbf{x}')| \geq b_2 t] \leq 2e^{-t\varepsilon^2/32}.$$

This proves the third property of  $T$  as listed above.  $\square$

**Contradiction.** To summarize, using the random map  $T$  together with the LSH family  $\mathcal{H}_B$  we are able to obtain a  $(r, cr, p_1', p_2')$ -sensitive family  $\mathcal{H}_H$  for  $D$ -dimensional Hamming space with  $p_1' = p_1 - \delta$  and  $p_2' = p_2 + \delta$  for  $\delta = 2e^{-t\varepsilon^2/32}$ . For our choice of  $c$  we plug the family  $\mathcal{H}_H$  into the lower bound of Lemma 5 and use that  $O(D^{-1/4}) = O(\varepsilon)$  which follows from our

constraint that  $\varepsilon \geq 1/d$ .

$$\begin{aligned}\rho(\mathcal{H}_H) &\geq \frac{1}{c} = \frac{\ln(1/(1 + \varepsilon/b_1)) + \ln(1/b_1)}{\ln(1/(1 - \varepsilon/b_2)) + \ln(1/b_2)} - O(\varepsilon) \\ &\geq \frac{\ln(1/b_1) - \varepsilon/b_1}{\ln(1/b_2) + 2\varepsilon/b_2} - O(\varepsilon) \\ &= \frac{\ln(1/b_1)}{\ln(1/b_2)} - O(\varepsilon)\end{aligned}$$

Under our assumed properties of  $\mathcal{H}_B$ , we can upper bound the value of  $\rho$  for  $\mathcal{H}_H$  where for simplicity we temporarily define  $\lambda = 2\delta/p_2$ . We assume that  $\lambda/\ln(1/p_2) \leq 1/2$  and  $\ln(1/p_2) \geq 1$  where the latter property holds without loss of generality through use of the standard LSH powering technique [20, 18, 27] that allows us to transform an LSH family with  $p_2 < 1$  (which we implicitly assume from the LSH definition) to a family that has  $p_2 \leq 1/e$  without changing its associated  $\rho$ -value.

$$\begin{aligned}\rho(\mathcal{H}_H) &= \frac{\ln(1/p_1')}{\ln(1/p_2')} = \frac{\ln(1/p_1) + \ln(1/(1 - \delta/p_1))}{\ln(1/p_2) + \ln(1/(1 + \delta/p_2))} \\ &\leq \frac{\ln(1/p_1) + \lambda}{\ln(1/p_2) - \lambda} = \frac{\ln(1/p_1) + \lambda}{(\ln 1/p_2)(1 - \lambda/(\ln 1/p_2))} \\ &\leq \frac{\ln(1/p_1) + \lambda}{\ln(1/p_2)} (1 + 2\lambda/(\ln 1/p_2)) = \frac{\ln(1/p_1)}{\ln(1/p_2)} + O(\delta/p_2) \\ &\leq \frac{\ln(1/b_1)}{\ln(1/b_2)} - \gamma + O(\delta/p_2).\end{aligned}$$

We get a contradiction between our upper bound and lower bound for  $\rho(\mathcal{H}_H)$  whenever  $\gamma$  violates the following relation that summarizes the bounds:

$$\frac{\ln(1/b_1)}{\ln(1/b_2)} - O(\varepsilon) \leq \rho(\mathcal{H}_H) \leq \frac{\ln(1/b_1)}{\ln(1/b_2)} - \gamma + O(\delta/p_2).$$

In order for a contradiction to occur, the value of  $\gamma$  has to satisfy

$$\gamma > O(\varepsilon) + O(\delta/p_2).$$

By our setting of  $t = \lfloor d/l \rfloor$  and  $l = \lceil 8/\varepsilon \rceil$  we have that  $\delta = e^{-\Omega(d\varepsilon^3)}$ . We can cause a contradiction for a setting of  $\varepsilon^3 = K \frac{\ln(d/p_2)}{d}$  where  $K$  is some constant and where we assume that  $d$  is greater than some constant. The value of  $\gamma$  for which the lower bound holds can be upper bounded by

$$\gamma = O\left(\frac{\ln(d/p_2)}{d}\right)^{1/3}.$$

This completes the proof of Theorem 3.

## 5 Conclusion and open problems

We have seen that, perhaps surprisingly, there exists a relatively simple way of strictly improving the  $\rho$ -value obtained by MinHash. To implement the required locality-sensitive map efficiently we introduce a new technique based on branching processes that could possibly lead to more efficient solutions in other settings.

It would be interesting to consider the possible time-space trade-offs. One approach to this would be to generalize the condition  $h_i(p \circ j) < \mathbf{x}_j/b_1|\mathbf{x}|$  to use different thresholds for queries and updates.

Another interesting question is if the improvement shown for sparse vectors can be achieved in general for inner product similarity. A similar, but possibly easier, direction would be to consider *weighted* Jaccard similarity.



**Acknowledgement.** We thank Thomas Dybdahl Ahle for comments on a previous version of this manuscript.

## A Equivalent set similarity search problems

**Regular similarity search problems.** In this section we consider how to use our data structure for Braun-Blanquet similarity search to support other similarity measures such as Jaccard similarity. We already observed in the introduction that a direct translation exists whenever the sizes of all sets are fixed to  $t$ . Call a similarity search problem  $(t, t')$ -regular if  $P$  is restricted to vectors of weight  $t$  and queries are restricted to vectors of weight  $t'$ . Obviously, a  $(t, t')$ -regular problem is no harder than the general similarity search problem, but it also cannot be too much easier: For every pair  $(t, t') \in \{0, \dots, d\}^2$  we can construct a  $(t, t')$ -regular data structure (such that each point  $\mathbf{x} \in P$  is represented in the  $d + 1$  data structures with  $t = |\mathbf{x}|$ ), and answer a query for  $\mathbf{q} \in \{0, 1\}^d$  by querying all data structures with  $t' = |\mathbf{q}|$ . Thus, the time and space for the general problem is at most  $d + 1$  times larger than the time and space of the most expensive  $(t, t')$ -regular data structure.

**Dimension reduction.** If the dimension is large a factor of  $d$  may be significant. However, for most natural similarity measures a  $(s_1, s_2)$ - $S$ -similarity problem in  $d \gg (\log n)^3$  dimensions can be reduced to a logarithmic number of  $(s'_1, s'_2)$ - $S$ -similarity problems on  $P' \subseteq \{0, 1\}^{d'}$  in  $d' = (\log n)^3$  dimensions with  $s'_1 = s_1 - O(1/\log n)$  and  $s'_2 = s_2 + O(1/\log n)$ . Since the similarity gap is close to the one in the original problem,  $s'_1 - s'_2 = s_1 - s_2 - O(1/\log n)$ , where  $s_1$  and  $s_2$  are assumed to be independent of  $n$ , the difficulty ( $\rho$ -value) remains essentially the same. First, split  $P$  into  $\log d$  size classes  $P_i$  such that vectors in class  $i$  have size in  $[2^i, 2^{i+1})$ . For each size class the reduction is done independently and works by a standard technique: sample a sequence of random sets  $I_j \subseteq \{1, \dots, d\}$ ,  $i = 1, \dots, d'$ , and set  $\mathbf{x}'_j = \bigvee_{\ell \in I_j} \mathbf{x}_\ell$ . The size of each set  $I_j$  is chosen such that  $\Pr[\mathbf{x}'_j = 1] \approx 1/\log(n)$  when  $|\mathbf{x}| = 2^{i+1}$ . By Chernoff bounds this mapping preserves the relative weight of vectors up to size  $2^i \log n$  up to an additive  $O(1/\log n)$  term with high probability. Assume now that the similarity measure  $S$  is such that for vectors in  $P_i$  we only need to consider  $|\mathbf{q}|$  in the range from  $2^i/\log n$  to  $2^i \log n$  (since if the size difference is larger, the similarity is negligible). Then we can apply Chernoff bounds to the relative weights of the dimension-reduced vectors  $\mathbf{x}'$ ,  $\mathbf{q}'$  and the intersection  $\mathbf{x}' \cap \mathbf{q}'$ . In particular, we get that the Jaccard similarity of a pair of vectors is preserved up to an additive error of  $O(1/\log n)$  with high probability. The class of similarity measures for which dimension reduction to  $(\log n)^{O(1)}$  dimensions is possible is large, and we do not attempt to characterize it here. Instead, we just note that for such similarity measures we can determine the complexity of similarity search up to a factor  $(\log n)^{O(1)}$  by only considering regular search problems.

**Equivalence of regular similarity search problems.** We call a set similarity measure on  $\{0, 1\}^d$  *symmetric* if it can be written in the form  $S(\mathbf{q}, \mathbf{x}) = f_{d, |\mathbf{q}|, |\mathbf{x}|}(|\mathbf{q} \cap \mathbf{x}|)$ , where each function  $f_{d, |\mathbf{q}|, |\mathbf{x}|}: \mathbb{N} \rightarrow [0, 1]$  is nondecreasing. All 59 set similarity measures listed in the survey [14], normalized to yield similarities in  $[0, 1]$ , are symmetric. In particular this is the case for Jaccard similarity (where  $J(\mathbf{q}, \mathbf{x}) = |\mathbf{q} \cap \mathbf{x}|/(|\mathbf{q}| + |\mathbf{x}| - |\mathbf{q} \cap \mathbf{x}|)$ ) and for Braun-Blanquet similarity. For a symmetric similarity measure  $S$ , the predicate  $S(\mathbf{q}, \mathbf{x}) \geq s_1$  is equivalent to the predicate  $|\mathbf{q} \cap \mathbf{x}| \geq i_1$ , where  $i_1 = \min\{i \mid f_{d, t', t}(i) \geq s_1\}$ , and  $S(\mathbf{q}, \mathbf{x}) > s_2$  is equivalent to the predicate  $|\mathbf{q} \cap \mathbf{x}| \geq i_2$ , where  $i_2 = \min\{i \mid f_{d, t', t}(i) > s_2\}$ . This means that every  $(t, t')$ -regular  $(s_1, s_2)$ - $S$ -similarity search problem on  $P \subseteq \{0, 1\}^d$  is equivalent to  $(i_1/d, i_2/d)$ - $I$ -similarity search problem on  $P$ , where  $I(\mathbf{q}, \mathbf{x}) = |\mathbf{x} \cap \mathbf{q}|/d$ . In other words, all regular similarity search problems can be translated to each other, and it suffices to study a single one, such as Braun-Blanquet similarity.

## B Details behind the lower bound

### B.1 Tools

For clarity we state some standard technical lemmas that we use to derive LSH lower bounds.

#### Hoeffding and Chernoff bounds.

**Lemma 7** (Hoeffding [19, Theorem 1]). *Let  $X_1, X_2, \dots, X_n$  be independent random variables satisfying  $0 \leq X_i \leq 1$  for  $i \in [n]$ . Define  $X = X_1 + X_2 + \dots + X_n$ ,  $Z = X/n$ , and  $\mu = \mathbb{E}[Z]$ , then:*

- For  $\hat{\mu} \geq \mu$  and  $0 < \varepsilon < 1 - \hat{\mu}$  we have that  $\Pr[Z - \hat{\mu} \geq \varepsilon] \leq e^{-2n\varepsilon^2}$ .
- For  $\hat{\mu} \leq \mu$  and  $0 < \varepsilon < \hat{\mu}$  we have that  $\Pr[Z - \hat{\mu} \leq -\varepsilon] \leq e^{-2n\varepsilon^2}$ .

**Lemma 8** (Chernoff [24, Theorems 4.4 and 4.5]). *Let  $X_1, \dots, X_n$  be independent Poisson trials and define  $X = \sum_{i=1}^n X_i$  and  $\mu = \mathbb{E}[X]$ . Then, for  $0 < \varepsilon < 1$  we have*

- $\Pr[X \geq (1 + \varepsilon)\mu] \leq e^{-\varepsilon^2\mu/3}$ .
- $\Pr[X \leq (1 - \varepsilon)\mu] \leq e^{-\varepsilon^2\mu/2}$ .

#### Bounding the natural logarithm and approximating the exponential function.

**Lemma 9** ([33]). *For  $x > -1$  we have that  $\frac{x}{1+x} \leq \ln(1+x) \leq x$ .*

**Lemma 10** ([26, Prop. B.3]). *For all  $t, n \in \mathbb{R}$  with  $|t| \leq n$  we have that  $e^t(1 - \frac{t^2}{n}) \leq (1 + \frac{t}{n})^n \leq e^t$ .*

### B.2 Proof of Lemma 5

**Preliminaries.** We will reuse the notation of Section 3.1 from O'Donnell et al. [27].

**Definition 4.** For  $0 \leq \lambda < 1$  we say that  $(\mathbf{x}, \mathbf{y})$  are  $(1 - \lambda)$ -correlated if  $\mathbf{x}$  is chosen uniformly at random from  $\{0, 1\}^d$  and  $\mathbf{y}$  is constructed by rerandomizing each bit from  $\mathbf{x}$  independently at random with probability  $\lambda$ .

Let  $(\mathbf{x}, \mathbf{y})$  be  $e^{-t}$ -correlated and let  $\mathcal{H}$  be a family of hash functions on  $\{0, 1\}^d$ , then we define

$$\mathbb{K}_{\mathcal{H}}(t) = \Pr_{\substack{h \sim \mathcal{H} \\ (\mathbf{x}, \mathbf{y}) \text{ } e^{-t}\text{-corr'd}}} [h(\mathbf{x}) = h(\mathbf{y})].$$

We have that  $\mathbb{K}_{\mathcal{H}}(t)$  is a log-convex function which implies the following property that underlies the lower bound:

**Lemma 11.** *For every family of hash functions  $\mathcal{H}$  on  $\{0, 1\}^d$ , every  $t \geq 0$ , and  $c \geq 1$  we have*

$$\frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t))}{\ln(1/\mathbb{K}_{\mathcal{H}}(ct))} \geq \frac{1}{c}. \quad (6)$$

The idea behind the proof is to tie  $p_1$  to  $\mathbb{K}_{\mathcal{H}}(t)$  and  $p_2$  to  $\mathbb{K}_{\mathcal{H}}(ct)$  through Chernoff bounds and then apply Lemma 11 to show that  $\rho \gtrsim 1/c$ .

**Proof.** Begin by assuming that we have a family  $\mathcal{H}$  that satisfies the conditions of Lemma 5. Note that the expected Hamming distance between  $(1 - \lambda)$ -correlated points  $\mathbf{x}$  and  $\mathbf{y}$  is given by  $(\lambda/2)d$ . We set  $\lambda_{p_1}/2 = d^{-1/2} - d^{-5/8}$  and  $\lambda_{p_2}/2 = cd^{-1/2} + 2cd^{-5/8}$  and let  $(\mathbf{x}, \mathbf{y})$  denote  $(1 - \lambda_{p_1})$ -correlated random strings and  $(\mathbf{x}, \mathbf{x}')$  denote  $(1 - \lambda_{p_2}q)$ -correlated random strings. By standard Chernoff bounds we get the following guarantees:

$$\begin{aligned}\Pr[\|\mathbf{x} - \mathbf{y}\|_1 \geq r] &\leq e^{-\Omega(d^{1/4})}, \\ \Pr[\|\mathbf{x} - \mathbf{x}'\|_1 \leq cr] &\leq e^{-\Omega(d^{1/4})}.\end{aligned}$$

We will establish a relationship between  $\mathbb{K}_{\mathcal{H}}(t_{p_1})$  and  $p_1$  on the one hand, and  $\mathbb{K}_{\mathcal{H}}(t_{p_2})$  and  $p_2$  on the other hand, for the following choice of parameters  $t_{p_1}$  and  $t_{p_2}$ :

$$\begin{aligned}t_{p_1} &= -\ln(1 - 2(d^{-1/2} - d^{-5/8})) \\ t_{p_2} &= -\ln(1 - 2c(d^{-1/2} + 2d^{-5/8})).\end{aligned}$$

By the properties of  $\mathcal{H}$  and from the definition of  $\mathbb{K}_{\mathcal{H}}$  we have that

$$\begin{aligned}\mathbb{K}_{\mathcal{H}}(t_{p_1}) &\geq p_1(1 - \Pr[\|\mathbf{x} - \mathbf{y}\|_1 > r]) \geq p_1 - \Pr[\|\mathbf{x} - \mathbf{y}\|_1 \geq r] \\ \mathbb{K}_{\mathcal{H}}(t_{p_2}) &\leq p_2(1 - \Pr[\|\mathbf{x} - \mathbf{x}'\|_1 \leq cr]) + \Pr[\|\mathbf{x} - \mathbf{x}'\|_1 \leq cr] \leq p_2 + \Pr[\|\mathbf{x} - \mathbf{x}'\|_1 \leq cr].\end{aligned}$$

Let  $\delta = \max\{\Pr[\|\mathbf{x} - \mathbf{y}\|_1 \geq r], \Pr[\|\mathbf{x} - \mathbf{x}'\|_1 \leq cr]\} = e^{-\Omega(d^{1/4})}$ . By Lemma 11 and our setting of  $t_{p_1}$  and  $t_{p_2}$  we can use the bounds on the natural logarithm from Lemma 9 to show the following:

$$\begin{aligned}\frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_1}))}{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))} &\geq \frac{t_{p_1}}{t_{p_2}} = \frac{\ln(1 - 2(d^{-1/2} - d^{-5/8}))}{\ln(1 - 2c(d^{-1/2} + 2d^{-5/8}))} \\ &\geq \frac{2(d^{-1/2} - d^{-5/8})}{2c(d^{-1/2} + 2d^{-5/8})} - 2(d^{-1/2} - d^{-5/8}) \\ &\geq \frac{1 - d^{-1/4}}{c + 2d^{-1/4}} - 2(d^{-1/2} - d^{-5/8}) \\ &= \frac{1}{c} - O(d^{-1/4}).\end{aligned}$$

We proceed by lower bounding  $\rho$  where we make use of the inequalities derived above.

$$\mathbb{K}_{\mathcal{H}}(t_{p_2}) - \delta \leq p_2 < p_1 \leq \mathbb{K}_{\mathcal{H}}(t_{p_1}) + \delta.$$

By Lemma 11 combined with the restrictions on our parameters, for  $d$  greater than some constant we have that  $\mathbb{K}_{\mathcal{H}}(t_{p_2}) \geq \mathbb{K}_{\mathcal{H}}(t_{p_1})^{2c} \geq (p_1/2)^{2c} \geq (2d)^{-2c} \geq (2d)^{-2d^{1/8}}$ . Furthermore, we lower bound  $\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))$  by using that  $\mathbb{K}_{\mathcal{H}}(t_{p_2}) \leq p_2 + \delta$  together with the restriction that  $p_2 \geq 1 - 1/d$  and the properties of  $\delta$ . For  $d$  greater than some constant it therefore holds that  $\mathbb{K}_{\mathcal{H}}(t_{p_2}) \leq 1 - 1/2d$  from which it follows that  $\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2})) \geq 1/2d$ .

$$\begin{aligned}\frac{\ln(1/p_1)}{\ln(1/p_2)} &\geq \frac{\ln(1/(\mathbb{K}_{\mathcal{H}}(t_{p_1}) + \delta))}{\ln(1/(\mathbb{K}_{\mathcal{H}}(t_{p_2}) - \delta))} \\ &= \frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_1})) - \ln(1 + \delta/\mathbb{K}_{\mathcal{H}}(t_{p_1}))}{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2})) + \ln(1/(1 - \delta/\mathbb{K}_{\mathcal{H}}(t_{p_2})))} \\ &\geq \frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_1})) - \delta/\mathbb{K}_{\mathcal{H}}(t_{p_1})}{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2})) + 2\delta/\mathbb{K}_{\mathcal{H}}(t_{p_2})} \\ &\geq \frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_1}))}{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))} - \frac{3\delta}{\mathbb{K}_{\mathcal{H}}(t_{p_2}) \ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))}.\end{aligned}$$

By the arguments above we have that

$$\frac{3\delta}{\mathbb{K}_{\mathcal{H}}(t_{p_2}) \ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))} = e^{-\Omega(d^{1/4})} = O(d^{-1/4}).$$

Inserting the lower bound for  $\frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_1}))}{\ln(1/\mathbb{K}_{\mathcal{H}}(t_{p_2}))}$  results in the lemma.

## C Comparisons

For completeness we state the proofs behind the comparisons between the  $\rho$ -values obtained by the CHOSEN PATH algorithm and other LSH techniques.

### C.1 Proof that $\rho < \rho_{\text{minhash}}$

For data sets with fixed sparsity and Braun-Blanquet similarities  $0 < b_2 < b_1 < 1$  we have that  $\rho/\rho_{\text{minhash}} = f(b_2)/f(b_1)$  where  $f(x) = \log(x/(2-x))/\log(x)$ . If  $f(x)$  is monotone increasing in  $(0;1)$  then  $\rho/\rho_{\text{minhash}} < 1$ . For  $x \in (0;1)$  we have that  $\text{sign}(f'(x)) = \text{sign}(g(x))$  where  $g(x) = \ln(x) + (2-x)\ln(2-x)$ . The function  $g(x)$  equals zero at  $x = 1$  and has the derivative  $g'(x) = \ln(x) - \ln(2-x)$  which is negative for values of  $x \in (0;1)$ . We can therefore see that  $f'(x)$  is positive in the interval and it follows that  $\rho < \rho_{\text{minhash}}$  for every choice of  $0 < b_2 < b_1 < 1$ .

### C.2 Proof that $\rho < \rho_{\text{angular}}$

We have that  $\rho/\rho_{\text{angular}} < 1$  if  $f(x) = \ln(x)\frac{1+x}{1-x}$  is a monotone increasing function for  $x \in (0;1)$ . For  $x \in (0;1)$  we have that  $\text{sign}(f'(x)) = \text{sign}(g(x))$  where  $g(x) = (1-x^2)/2 + x \ln x$ . We note that  $g(1) = 0$  and  $g'(x) = 1 - x + \ln x$ . Therefore, if  $g'(x) < 0$  for  $x \in (0;1)$  it holds that  $g(x) > 0$  and  $f(x)$  is monotone increasing in the same interval. We have that  $g'(1) = 0$  and  $g''(x) = -1 + 1/x > 0$  implying that  $g'(x) < 0$  in the interval.

### C.3 Comparing $\rho$ and $\rho_{\text{datadep}}$

**Lemma 12.** *Let  $0 < b_2 < b_1 < 1$  and fix  $\rho = 1/2$  such that  $b_1 = \sqrt{b_2}$ . Then we have that  $\rho < \rho_{\text{datadep}}$  for every value of  $b_2 < 1/4$ .*

*Proof.* We will compare  $\rho = \log(b_1)/\log(b_2)$  and  $\rho_{\text{datadep}} = \frac{1-b_1}{1+b_1-2b_2}$  when  $\rho$  is fixed at  $\rho = 1/2$ , or equivalently,  $b_1 = \sqrt{b_2}$ . We can solve the quadratic equation  $1/2 = \frac{1-\sqrt{b_2}}{1+\sqrt{b_2}-2b_2}$  to see that for  $0 < b_2 < 1$  we have that  $\rho = \rho_{\text{datadep}}$  only when  $b_2 = 1/4$ . The derivative of  $\rho_{\text{datadep}}$  with respect to  $b_2$  is negative when  $b_1 = \sqrt{b_2}$ . Under this restriction we therefore have that  $\rho < \rho_{\text{datadep}}$  for  $b_2 < 1/4$  which is equivalent to  $j_2 < 1/7$  in the fixed-weight setting.  $\square$

To compare  $\rho$ -values over the full parameter space we use the following two Lemmas.

**Lemma 13.** *For every choice of fixed  $0 < \rho < 1$  let  $b_2 = b_1^{1/\rho}$ . Then  $\rho_{\text{datadep}} = \frac{1-b_1}{1+b_1-2b_2}$  is decreasing in  $b_1$  for  $b_1 \in (0;1)$ .*

*Proof.* The sign of the derivative of  $\rho_{\text{datadep}}$  with respect to  $b_1$  is equal to the sign of the function  $g(x) = -\rho x^{-1/\rho} + \rho - 1 + x^{-1}$  for  $x \in (0;1)$ . We have that  $g(1) = 0$  and  $g'(x) = x^{-1/\rho} - 1 - x^{-2} > 0$  for  $x \in (0;1)$  which shows that  $g(x) < 0$  in the interval.  $\square$

**Lemma 14.** *For  $1/5 = b_2 < b_1 < 1$  we have that  $\rho < \rho_{\text{datadep}}$ .*

*Proof.* For fixed  $b_2 = 1/5$  consider  $f(b_1) = \rho - \rho_{\text{datadep}}$  as a function of  $b_1$  in the interval  $[1/5, 1]$ . We want to show that  $f(b_1) < 0$  for  $b_1 \in (1/5; 1)$ . In the endpoints the function takes the value 0. Between the endpoints we find that  $f'(b_1) = \frac{1}{\ln(5)b_1} + \frac{8/5}{(3/5+b_1)^2}$  and that  $f'(b_1) = 0$  is a quadratic form with only one solution  $b_1^*$  in  $[1/5; 1]$ . By Lemma 12 we know that that for  $b_2 = 1/5$  and  $b_1 = 1/\sqrt{5}$  it holds that  $f(b_1) < 0$ . Since  $f(1/5) = f(1) = 0$ ,  $f'(b_1) = 0$  only in a single point in  $[1/5; 1]$ , and  $f(1/\sqrt{5}) < 0$  we can conclude that the Lemma holds.  $\square$

**Corollary 2.** *For every choice of  $b_1, b_2$  satisfying  $0 < b_2 \leq 1/5$  and  $b_2 < b_1 < 1$  we have that  $\rho < \rho_{\text{datadep}}$ .*

*Proof.* If  $b_2 = 1/5$  the property holds by Lemma 14. If  $b_2 < 1/5$  we define new variables  $\hat{b}_2, \hat{b}_1$ , setting  $\hat{b}_1 = \hat{b}_1^{\rho(b_1, b_2)}$  and initially consider  $\hat{b}_2 = 1/5$ . In this setting we again have that  $\rho(\hat{b}_1, \hat{b}_2) < \rho_{\text{datadep}}(\hat{b}_1, \hat{b}_2)$ . According to Lemma 13 it holds that  $\rho_{\text{datadep}}$  is decreasing in  $b_2$  for fixed  $\rho$ . Therefore, as  $\hat{b}_2$  decreases to  $\hat{b}_2 = b_2$  where  $\hat{b}_1 = b_1$  we have that  $\rho(\hat{b}_1, \hat{b}_2) = \rho$  remains constant while  $\rho_{\text{datadep}}$  increases. Since it held that  $\rho < \rho_{\text{datadep}}$  at the initial values of  $\hat{b}_1, \hat{b}_2$  it must also hold for  $b_1, b_2$ .  $\square$

**Numerical comparison of  $\rho_{\text{minhash}}$  and  $\rho_{\text{datadep}}$ .** Comparing  $\rho_{\text{minhash}}$  to  $\rho_{\text{datadep}}$  we can verify numerically that even for  $b_2$  fixed as low as  $b_2 = 1/23$  we can find values of  $b_1$  (for example  $b_1 = 0.995$  such that  $\rho_{\text{minhash}} > \rho_{\text{datadep}}$ .

## References

- [1] T. D. Ahle, R. Pagh, I. P. Razenshteyn, and F. Silvestri. On the complexity of inner product similarity join. In *Proceedings of the 35th Symposium on Principles of Database Systems (PODS)*. ACM, 2016.
- [2] A. Andoni, P. Indyk, T. Laarhoven, I. Razenshteyn, and L. Schmidt. Practical and optimal lsh for angular distance. In *Proc. NIPS '15*, pages 1225–1233, 2015.
- [3] A. Andoni, P. Indyk, H. L. Nguyen, and I. P. Razenshteyn. Beyond locality-sensitive hashing. In *Proc. SODA '14*, pages 1018–1028, 2014.
- [4] A. Andoni, T. Laarhoven, I. P. Razenshteyn, and E. Waingarten. Optimal hashing-based time-space trade-offs for approximate near neighbors. *CoRR*, abs/1608.03580, 2016. Accepted for publication in SODA '17.
- [5] A. Andoni and I. Razenshteyn. Optimal data-dependent hashing for approximate near neighbors. In *Proc. STOC '15*, pages 793–801, 2015.
- [6] A. Andoni and I. Razenshteyn. Tight lower bounds for data-dependent locality-sensitive hashing. In *Proc. SoCG '16*, pages 9:1–9:11, 2016.
- [7] A. Arasu, V. Ganti, and R. Kaushik. Efficient exact set-similarity joins. In *Proceedings of the 32nd international conference on Very large data bases*, pages 918–929. VLDB Endowment, 2006.
- [8] R. J. Bayardo, Y. Ma, and R. Srikant. Scaling up all pairs similarity search. In *Proceedings of the 16th international conference on World Wide Web*, pages 131–140. ACM, 2007.
- [9] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *Proc. SODA '16*, pages 10–24, 2016.
- [10] J. Braun-Blanquet. *Plant sociology. The study of plant communities*. McGraw-Hill, 1932.

- [11] A. Z. Broder. On the resemblance and containment of documents. In *Compression and Complexity of Sequences 1997. Proceedings*, pages 21–29. IEEE, 1997.
- [12] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig. Syntactic clustering of the web. *Computer Networks and ISDN Systems*, 29(8):1157–1166, 1997.
- [13] M. Charikar. Similarity estimation techniques from rounding algorithms. In *Proc. STOC '02*, pages 380–388, 2002.
- [14] S. Choi, S. Cha, and C. C. Tappert. A survey of binary similarity and distance measures. *J. Syst. Cybern. Informatics*, 8(1):43–48, 2010.
- [15] T. Christiani. A framework for similarity search with space-time tradeoffs using locality-sensitive filtering. *CoRR*, abs/1605.02687, 2016. Accepted for publication at SODA’17.
- [16] M. Dubiner. Bucketing coding and information theory for the statistical high-dimensional nearest-neighbor problem. *IEEE Trans. Information Theory*, 56(8):4166–4179, 2010.
- [17] T. Hagerup. Sorting and searching on the word RAM. In *Proc. STACS '98*, pages 366–398, 1998.
- [18] S. Har-Peled, P. Indyk, and R. Motwani. Approximate nearest neighbor: Towards removing the curse of dimensionality. *Theory of computing*, 8(1):321–350, 2012.
- [19] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Jour. Am. Stat. Assoc.*, 58(301):13–30, 1963.
- [20] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proc. STOC '98*, pages 604–613, 1998.
- [21] T. Laarhoven. Tradeoffs for nearest neighbors on the sphere. *CoRR*, abs/1511.07527, 2015.
- [22] P. Li and A. C. König. Theory and applications of b-bit minwise hashing. *Communications of the ACM*, 54(8):101–109, 2011.
- [23] M. Mitzenmacher, R. Pagh, and N. Pham. Efficient estimation for high similarities using odd sketches. In *Proc. WWW '14*, pages 109–118, 2014.
- [24] M. Mitzenmacher and E. Upfal. *Probability and computing*. Cambridge University Press, New York, NY, 2005.
- [25] R. Motwani, A. Naor, and R. Panigrahy. Lower bounds on locality sensitive hashing. *SIAM J. Discrete Math.*, 21(4):930–935, 2007.
- [26] R. Motwani and P. Raghavan. *Randomized algorithms*. Chapman & Hall/CRC, 2010.
- [27] R. O’Donnell, Y. Wu, and Y. Zhou. Optimal lower bounds for locality-sensitive hashing (except when  $q$  is tiny). *ACM Transactions on Computation Theory (TOCT)*, 6(1):5, 2014.
- [28] R. Pagh, M. Stöckel, and D. P. Woodruff. Is min-wise hashing optimal for summarizing set intersection? In *Proceedings of the 33rd ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 109–120. ACM, 2014.
- [29] R. Panigrahy, K. Talwar, and U. Wieder. Lower bounds on near neighbor search via metric expansion. In *Proc. FOCS '10*, pages 805–814, 2010.
- [30] A. Shrivastava and P. Li. Asymmetric minwise hashing for indexing binary inner products and set containment. In *Proceedings of the 24th International Conference on World Wide Web*, pages 981–991. ACM, 2015.

- [31] K. Terasawa and Y. Tanaka. Spherical LSH for approximate nearest neighbor search on unit hypersphere. In *Proc. WADS '07*, pages 27–38, 2007.
- [32] M. Thorup. Bottom-k and priority sampling, set similarity and subset sums with minimal independence. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 371–380. ACM, 2013.
- [33] F. Topsøe. *Some Bounds for the Logarithmic Function*, volume 4, pages 137–151. Nova Science, 2007.
- [34] A. L. Zobrist. A new hashing method with application for game playing. *ICCA journal*, 13(2):69–73, 1970.